Cx	Secret Key of Entity **X**
Dx	Private Key of Entity **X** (a pair **dx**, **nx**)
dx	Private Exponent of **Dx**
Ex	Public Key of Entity **X** (a pair **ex**, **nx**)
ex	Public Exponent of **Ex**
K	Any cryptographic key, Symmetric Key
Ko	Group Symmetric Key
Koo	Master Symmetric Key
K{M}	The Encryption Function of Message **M** using the Key **K**
Kxy	Session Key, Common Secret Key between **X** and **Y**
Lx	License or Certificate issued to **X**
M	Plain Message, Plaintext
Mx	Message to or from Entity **X**
Nx	ID # of Entity **X**
Ni	ID # of User **I**
Nj	ID # of System Terminal **J**
nx	Modulus of the key pair **Dx**, **Ex**
O	System Authority
P	Encrypted Message, Cipher Message, Ciphertext
PWx	Password of **X**
Qx	Challenge Question, Random Number sent to **X**
Rx	Response, Signed by **X**
Sx	Message Signed by **X**
X	Unknown Entity
Y	Unknown Entity (Authenticator)
Z	Unknown Entity (Authenticatee)

FIG. 1: Notation

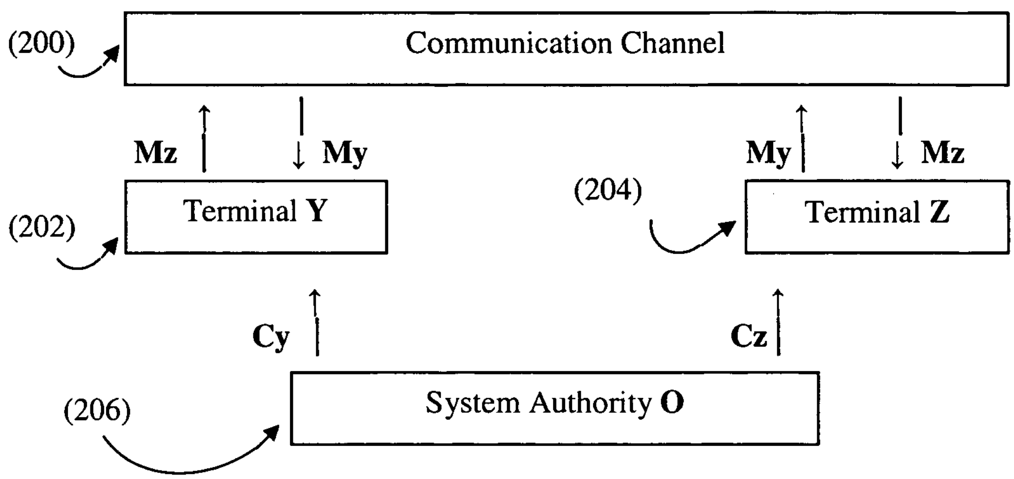
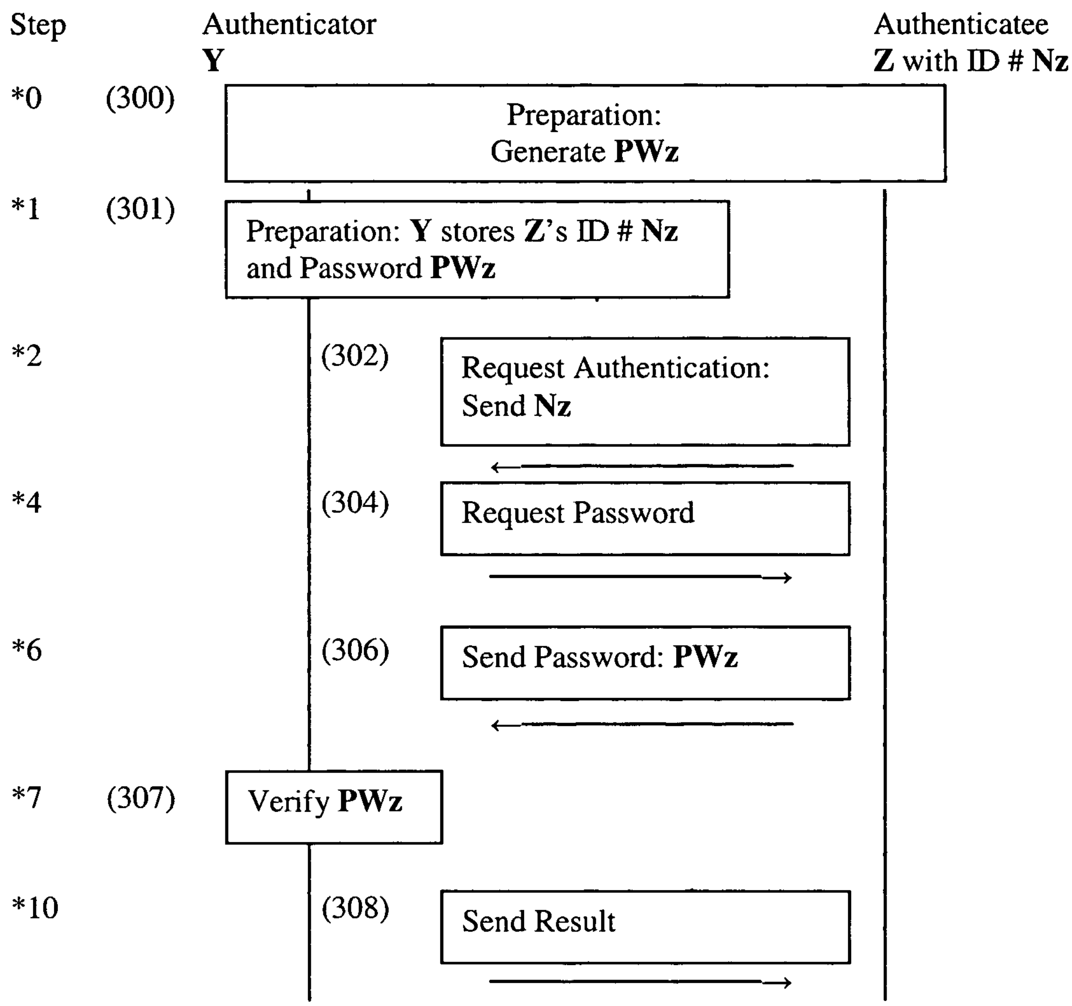


FIG. 2: Block Diagram of this Invention, S-RSA



where
Y : Authenticator
Z : Authenticatee
Nz : ID # of **Z**
PWz : Password of **Z**

FIG. 3: Flow of Conventional Password Authentication

Encrypt

$$\boxed{\mathbf{P = K\,\{M\}}} \tag{402}$$

M is encrypted by **K**

Decrypt

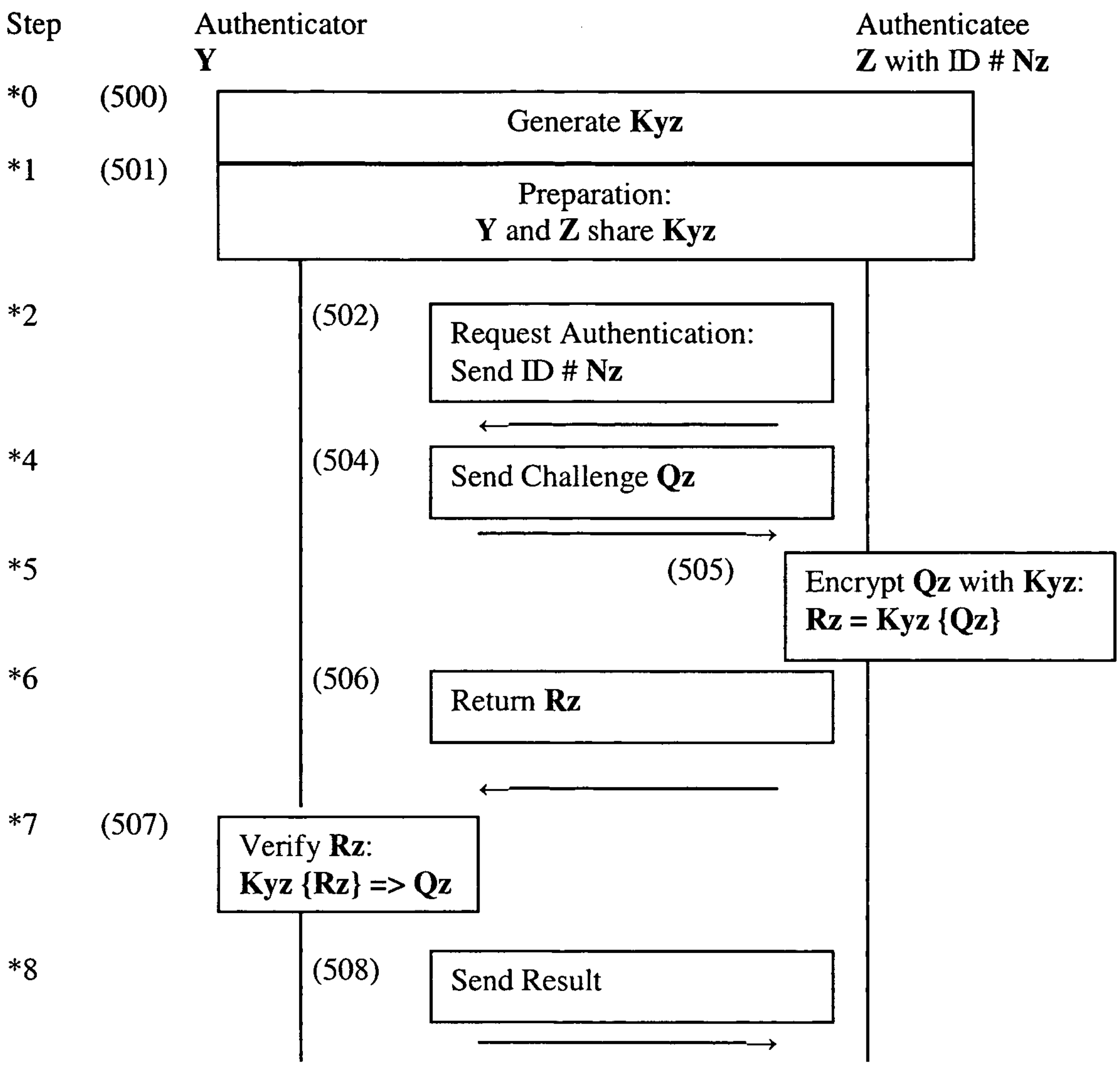
$$\boxed{\mathbf{M = K\,\{P\}}} \tag{404}$$

P is decrypted by **K**

where

P : Ciphertext
K : Symmetric Key
M : Plaintext
{ } : Cryptographic Function

FIG. 4: Formulae of Symmetric Key Encryption



where

Y : Authenticator
Z : Authenticatee
Nz : ID # of **Z**
Kyz : Secret Common Key between **Y** and **Z**
Qz : Challenge Message, Random Number sent to **Z**
Rz : Response Message from **Z**

FIG. 5: Flow of Conventional Symmetric Key Authentication

Encrypt

$$\boxed{\begin{aligned} \mathbf{P} &= \mathbf{E}\,\{\mathbf{M}\} \\ &= \mathbf{M}^{e}\,(\text{mod } \mathbf{n}) \end{aligned}} \qquad (602)$$

M is encrypted by **E**

Decrypt

$$\boxed{\begin{aligned} \mathbf{M} &= \mathbf{D}\,\{\mathbf{P}\} \\ &= \mathbf{P}^{d}\,(\text{mod } \mathbf{n}) \\ &= \mathbf{M}^{e*d}\,(\text{mod } \mathbf{n}) \\ &= \mathbf{M} \end{aligned}} \qquad (604)$$

P is decrypted by **D**

Sign

$$\boxed{\mathbf{S} = \mathbf{D}\,\{\mathbf{M}\}} \qquad (606)$$

M is signed by **D**

Verify

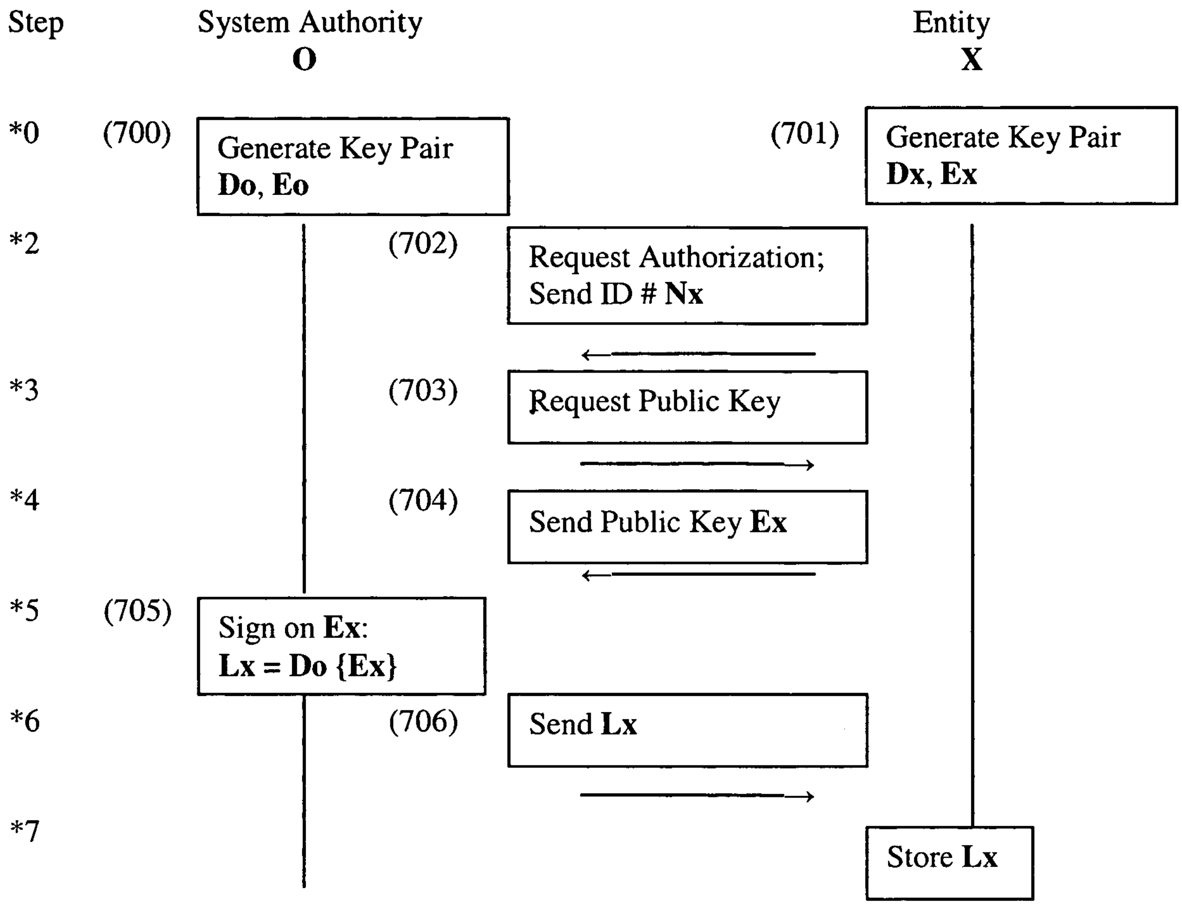
$$\boxed{\mathbf{E}\,\{\mathbf{S}\} \Rightarrow \mathbf{M}} \qquad (608)$$

S is verified by **E**

where

P	:	Ciphertext
E	:	Public Key (pair **e**, **n**)
D	:	Private Key (pair **d**, **n**)
n	:	Modulus of Key pair **E**, **D**
M	:	Plaintext
S	:	Signed Message
{ }	:	Cryptographic Function

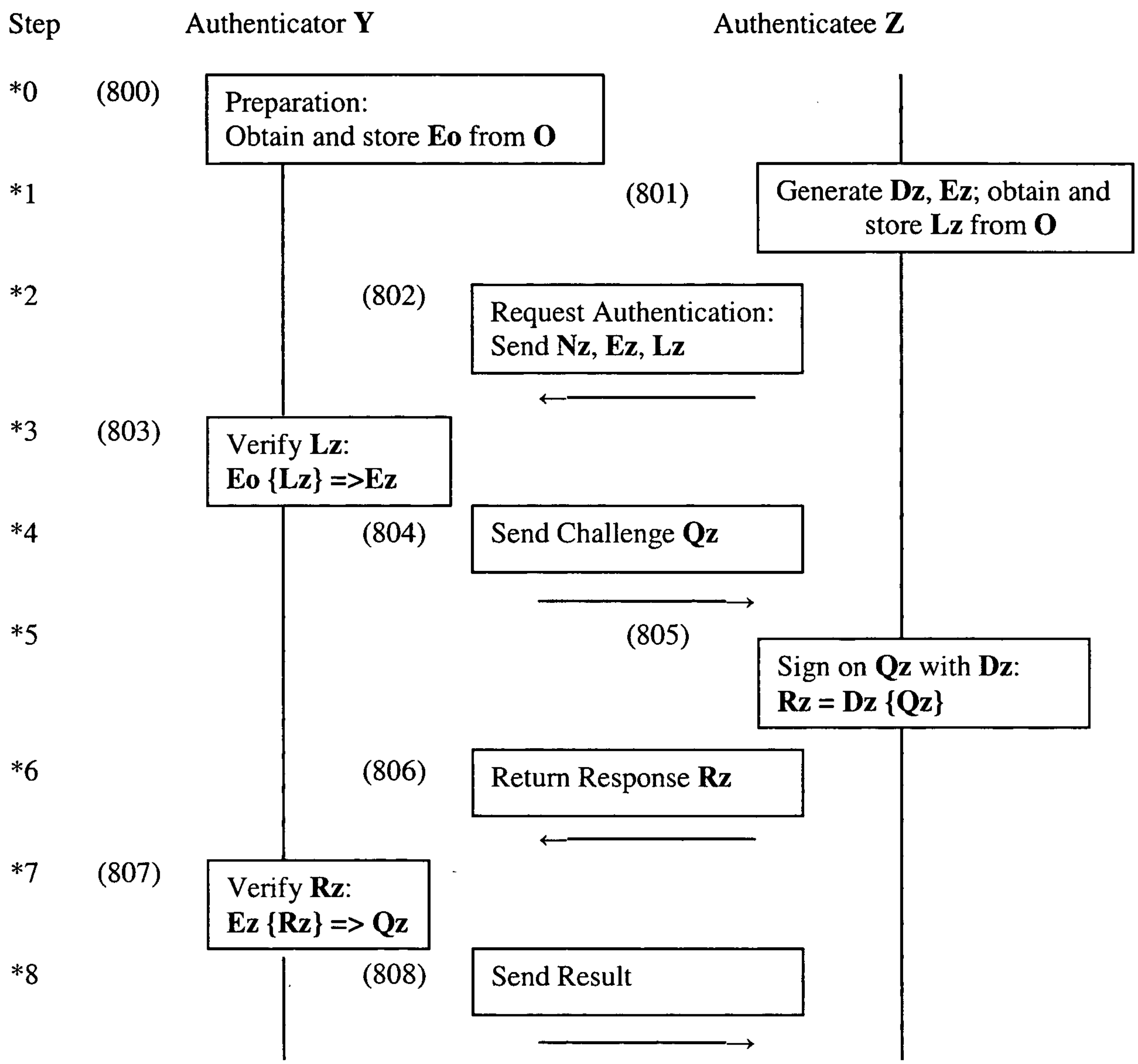
FIG. 6: Standard Formulae of RSA



where

Nx : ID # of **X**
Do : Private Key of System Authority **O**
Eo : Public Key of System Authority **O**
Dx : Private Key of Entity **X**
Ex : Public Key of Entity **X**
Lx : Certificate issued to **X**

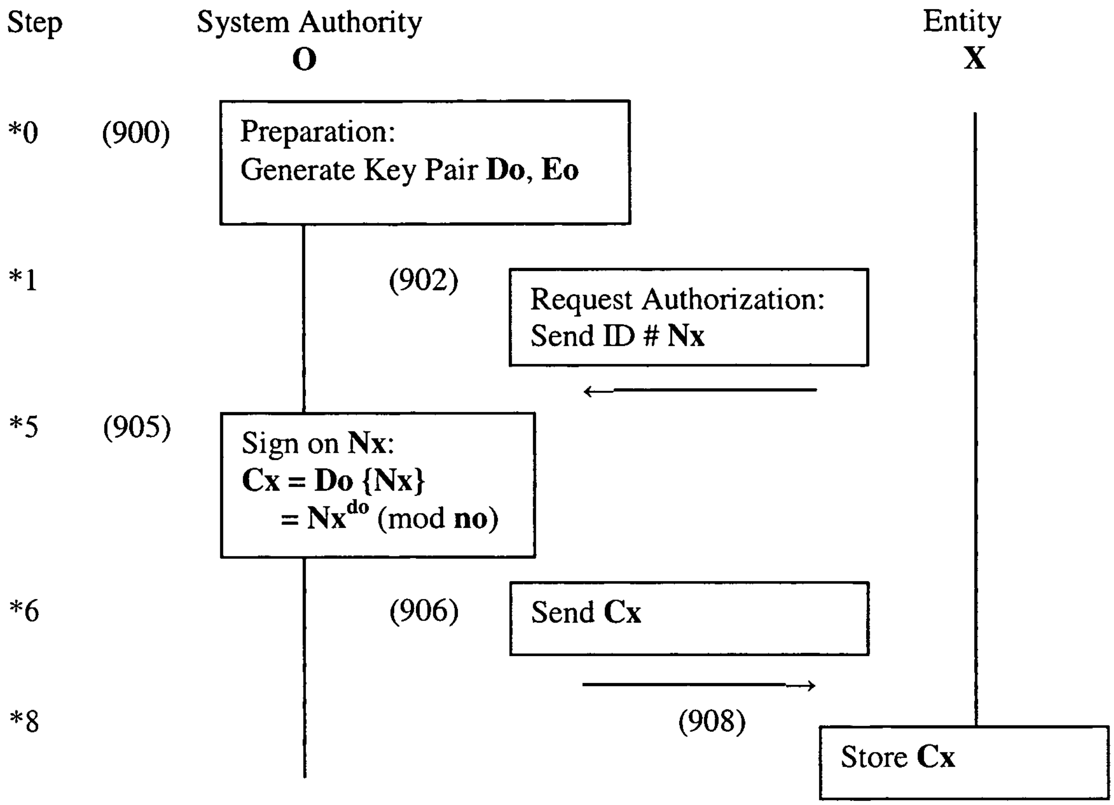
FIG. 7: Preparation Flow of RSA



where

Eo : Public Key of System Authority **O**
Dz : Private Key of **Z**
Ez : Public Key of **Z**
Lz : Certificate issued to **Z**
Qz : Challenge Message, Random Number sent to **Z**
Rz : Response from **Z**, Signed Message

FIG. 8: Flow of Regular RSA Key Authentication



where

Nx : ID # of **X**
Do : Private Key of System Authority **O**
Eo : Public Key of System Authority **O**
do : Private Exponent
no : Modulus of key pair **Do, Eo**
Cx : Secret Key of **X**

FIG. 9: Preparation Flow of This Invention, S-RSA

Sign

$$\boxed{\begin{aligned}\mathbf{Sx} &= \mathbf{Mx}\,\{\mathbf{Cx}\} \\ &= \mathbf{Cx}^{\mathbf{Mx}} \text{ (mod } \mathbf{no}\text{)}\end{aligned}} \tag{1006}$$

Verify

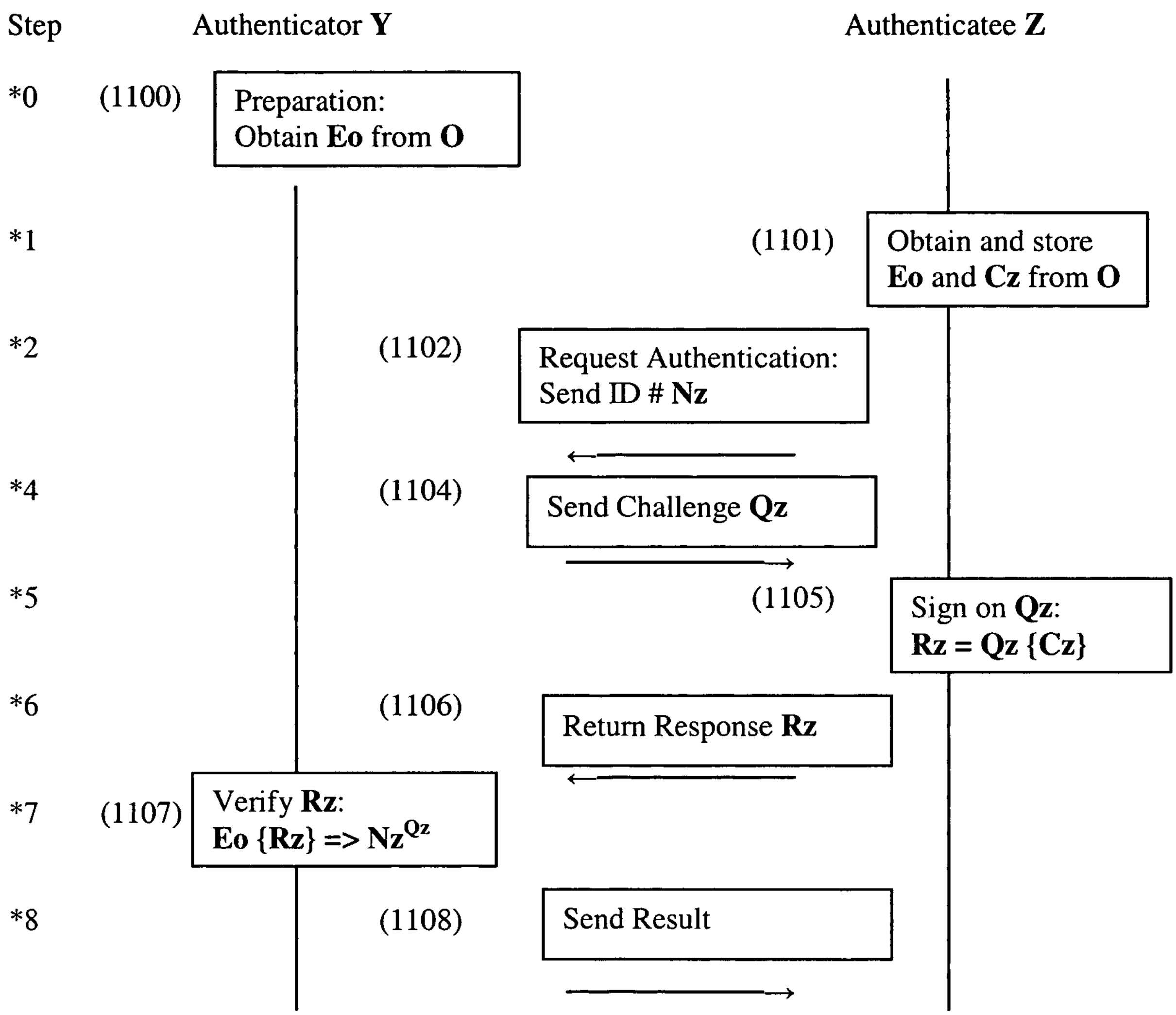
$$\boxed{\begin{aligned}&\mathbf{Eo}\,\{\mathbf{Sx}\} \\ &= \mathbf{Sx}^{\mathbf{eo}} \text{ (mod } \mathbf{no}\text{)} \\ &= \mathbf{Cx}^{\mathbf{Mx*eo}} \text{ (mod } \mathbf{no}\text{)} \\ &= \mathbf{Nx}^{\mathbf{do*Mx*eo}} \text{ (mod } \mathbf{no}\text{)} \\ &= \mathbf{Nx}^{\mathbf{Mx}} \text{ (mod } \mathbf{no}\text{))}\end{aligned}} \tag{1008}$$

Since $\mathbf{Nx}^{\mathbf{do*eo}}$ (mod $\mathbf{no}$) = $\mathbf{Nx}$

where

Nx : ID # of **X** or License # issued to **X**
Do : Private Key of System Authority **O**
do : Private Exponent
Eo : Public Key of System Authority **O**
eo : Public Exponent
no : Modulus of key pair **Do, Eo**
Cx : Secret Key of **X** where $\mathbf{Cx} = \mathbf{Nx}^{\mathbf{do}}$ (mod $\mathbf{no}$)
Mx : Message of **X**
Sx : Message Signed by **X**

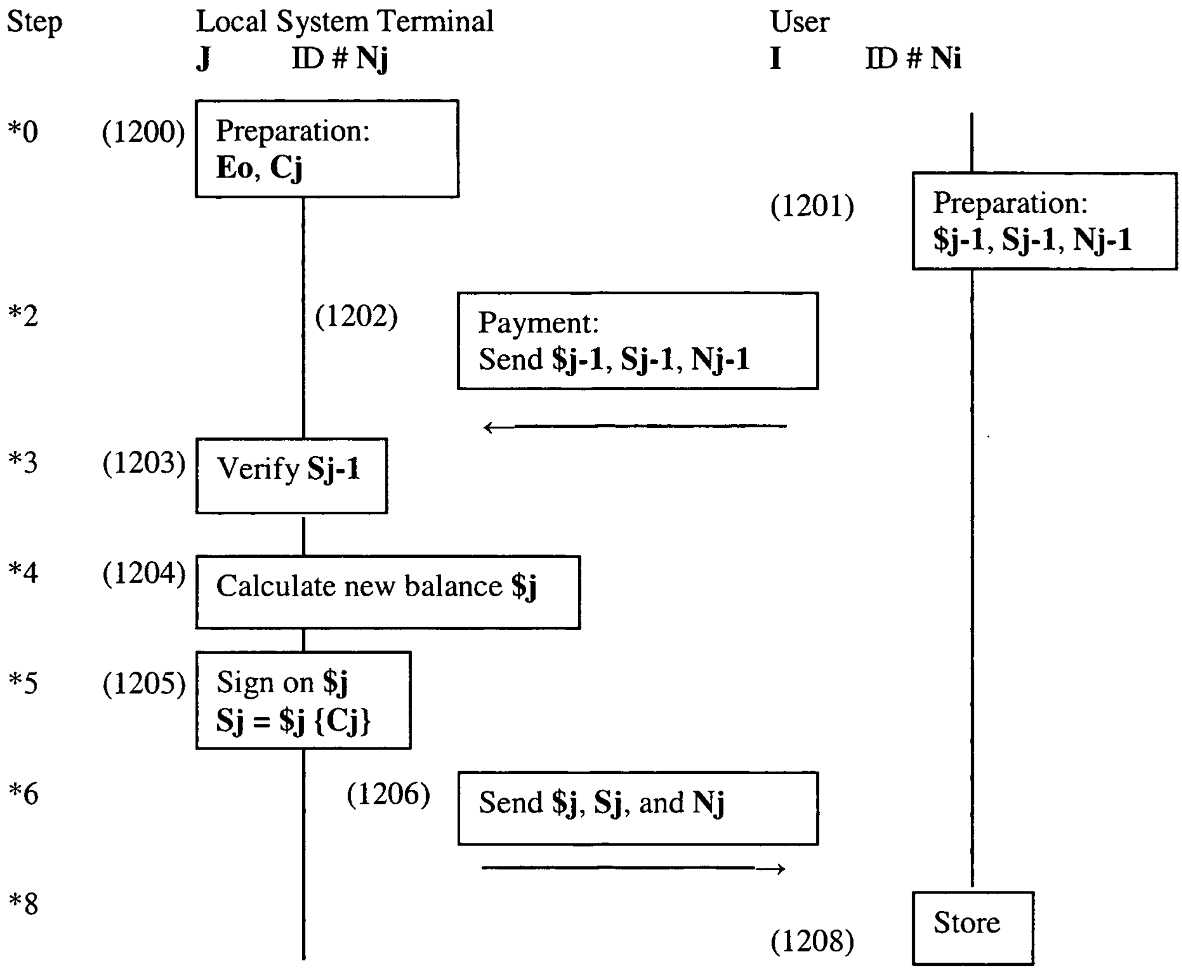
FIG. 10: Signing Formulae of This Invention, S-RSA



where

Nz : ID # of **Z**, or License # issued to **Z**
Eo : Public Key of System Authority **O**
Cz : Secret Key of **Z**
Qz : Challenge Message, Random Number sent to **Z**
Rz : Response from **Z**, Signed Message

FIG. 11: Authentication Flow of This Invention, S-RSA



where

Nj : ID # of Local System Terminal **J**
Nj-1 : ID # of Most Recently Visited Terminal **j-1**
Eo : Public Key of System Authority **O**
Cj : Secret Key of Terminal **J**
$j-1 : Present Balance received from Most Recently Visited Terminal **j-1**
$J : New Balance
Sj-1 : Present Balance signed by **J-1**
Sj : New Balance signed by **J**

FIG. 12: Signing Payment Flow of This Invention, S-RSA

$$\boxed{\begin{aligned}\mathbf{Pz} &= \mathbf{Ey}\ \{\mathbf{Mz}\} \\ &= \mathbf{Mz}^{ey}\ (\text{mod}\ \mathbf{ny})\end{aligned}} \qquad (1302)$$

Z sends message **Mz** to **Y**, wrapping it with **Y**'s public key **Ey**

where
Y : Authenticator
Z : Authenticatee
Ey : Public Key of Entity **Y**
ey : Public Exponent
ny : Modulus of **Y**'s Public Key
Mz : Message of **Z**
Pz : Encrypted Message of **Z**

$$\boxed{\mathbf{P} = \mathbf{M}^{e}\ (\text{mod}\ \mathbf{n})} \qquad (1304)$$

$$\boxed{\begin{aligned}\mathbf{P} &= (\mathbf{M}^2)^{16}\ {*}(\mathbf{M})\ (\text{mod}\ \mathbf{n}) \\ &= (\mathbf{M}^2)^2 \ldots)^2 * (\mathbf{M})\ (\text{mod}\ \mathbf{n}) \\ &\quad \text{since } \mathbf{E} = \mathbf{2}^{16} + \mathbf{1}\end{aligned}} \qquad (1306)$$

Multiplicative and modular operations must be repeated 17 times

where
E : Public Key
n : Modulus of Public Key
M : Plain Message
P : Encrypted Message

FIG. 13: Secure Socket Layer Communication

If **Qz** is a 16 bit number

and $\mathbf{Qz} = \mathbf{2}^{15 * b15 + 14 * b14 + \ldots + 1 * b1 + 0 * b0}$

where **bi** = **0** or **1**, then

$$\boxed{\begin{array}{l} \mathbf{Qz}\,\{\mathbf{Cz}\} \\ = (\mathbf{Cz}^2)^{15 * b15} * (\mathbf{Cz}^2)^{14 * b14} * \ldots * (\mathbf{Cz}^2)^{1 * b1} * (\mathbf{Cz})^{b0} \pmod{\mathbf{No}} \\ \text{if } \mathbf{bi} = 0. \\ (\mathbf{Cz}^2)^{i * bi} = 1 \end{array}}$$

(1402)

Therefore, if a table of $(\mathbf{Cz}^2)^i$ is pre-calculated, only eight multiplicative and modular operations must be performed on average.

The table size is

16 x 1024 bit = 2KB

(1404)

FIG. 14: Calculation Time of This Invention, S-RSA

Cz $(\mathbf{Cz}^2)^1$ (mod **no**) $(\mathbf{Cz}^2)^2$ (mod **no**) $(\mathbf{Cz}^2)^3$ (mod **no**) $(\mathbf{Cz}^2)^{15}$ (mod **no**)	x x x.....x x x x x x x x.....x x x x x
2 Bytes	1024 bit

Total 32 Bytes + 2 KBytes

FIG. 15: Table of Powers of **Cz**